Activer l'API Docker sur l'instance "web-srv"

Activer TLS pour l'API Docker

- Avec les versions récentes de Docker (décembre 2022), il devient nécessaire d'activer TLS sur l'API Docker. Voici la procédure.
- Créer le dossier dans la config de Docker qui contiendra le certificat et les fichiers associés : `mkdir /etc/docker/ssl && chmod 700 /etc/docker/ssl/ && cd /etc/docker/ssl`
- Vous pouvez suivre la démarche indiquée dans la documentation de Docker pour générer le certificat et les fichiers associés (même ceux nécessaire au client) sur le serveur où l'API doit être exposée. Vous pouvez comparer à la démache retranscrite ici appliqué à notre configuration sur le serveur d'IP privée 10.0.1.10 (source :
  - Generate CA, of course, keep same CA for each Docker API certs you want to generate on other machines:
    - `openssl genrsa -out ca-key.pem 4096`
    - CA valable 1 an (365 ⇒ augmenter à 1093 ?) : `openssl req -new -x509 -days 365 -key ca-key.pem -sha256 -out ca.pem`
  - Generate certs for our 10.0.1.10 Docker API. This line is just descriptive, it will work for another IP or domain :
    - `openssl genrsa -out server-key.pem 4096`
    - `openssl req -subj "/CN=10.0.1.10" -sha256 -new -key server-key.pem -out server.csr`
  - Allow external connections for specific IPs and hosts. Client requesting the API MUST MATCH the following subjectAltNames :
    - `echo "subjectAltName=IP:10.0.1.10,IP:127.0.0.1" > extfile.cnf`
    - Signature du certificat avec le CA valable 1 an (365 ⇒ augmenter à 1093 ?) : `openssl x509 -req -days 365 -sha256 -in server.csr -CA ca.pem -CAkey ca-key.pem -CAcreateserial -out server-cert.pem -extfile extfile.cnf '' * Client certificates (for instance, might be used for Portainer) : * ''openssl genrsa -out key.pem 4096`
    - `openssl req -subj '/CN=client' -new -key key.pem -out client.csr`
    - `echo extendedKeyUsage = clientAuth > extfile-client.cnf`
    - valable 1 an (365 ⇒ augmenter à 1093 ?) : `openssl x509 -req -days 365 -sha256 -in client.csr -CA ca.pem -CAkey ca-key.pem -CAcreateserial -out cert.pem -extfile extfile-client.cnf`
  - Removing files that won't be used anymore :
    - `rm -v client.csr server.csr extfile.cnf extfile-client.cnf`
  - Setting correct rights for keys :
    - `chmod -v 0400 ca-key.pem key.pem server-key.pem`
    - `chmod -v 0444 ca.pem server-cert.pem cert.pem`
- Créer ou éditer le fichier '' /etc/docker/daemon.json '' avec : ''vi /etc/docker/daemon.json ''
  - Ajouter le contenu suivant:

```
{
    "tls": true,
    "tlsverify": true,
```

```
    "tlscacert": "/etc/docker/ssl/ca.pem",
    "tlscert": "/etc/docker/ssl/server-cert.pem",
    "tlskey": "/etc/docker/ssl/server-key.pem"
}
```

- Vous devez ensuite activer la persistance de l'API Docker comme indiqué ci-dessous

Activer avec persistance l'API Docker

- Afin d'éviter que les modifications effectuées dans le fichier */lib/systemd/system/docker.service* soient écrasées à chaque mise à jour de Docker, vous devez ajouter un fichier qui écrasera les valeurs par défaut.
    - **Source** : Using systemd to control the Docker daemon
- Pour créer automatiquement l'arborescence de dossier et le fichier nécessaire, utiliser la commande suivante : '' systemctl edit docker ''
    - La commande précédente ouvre l'éditeur par défaut du système, vous pouvez ajouter le contenu suivant et sortir de l'édition du fichier en sauvegardant :

```
[Service]
ExecStart=
ExecStart=/usr/bin/dockerd -H fd:// --
containerd=/run/containerd/containerd.sock -H tcp://10.0.1.10:2376
```

        - **Note** : la première ligne *ExecStart=* vide permet de réinitialiser la commande de lancement de Docker
    - Les modifications devraient être présente dans le fichier suivant : '' vi /etc/systemd/system/docker.service.d/override.conf ''
- Lancer la prise en compte des modifications qui vérifiera une éventuelle erreur : '' systemctl daemon-reload ''
- Relancer le service Docker : '' systemctl restart docker ''
- Vérifier la présence des nouveaux paramètres dans *CGroup* : '' systemctl status docker ''

Tester l'activation de l'API

- Si **vous n'avez pas encore activer la permanence de l'accès à l'API** sur le serveur *web-srv*, vous pouvez activer temporaire l'API *Docker* sur l'IP de l'hôte du VPN : `vi /lib/systemd/system/docker.service`
    - Modifier la ligne `ExectStart=` en ajoutant l'option `-H tcp://10.0.1.20:2376` juste aprés `-H fd://`
    - Prendre en compte les changements : `systemctl daemon-reload`
    - Redémarrer Docker : `systemctl restart docker`
- Pour tester depuis l'hôte hébergeant Portainer :
    - récupérer les fichiers `key.pem` et `cert.pem` généré précédemment et stockés sur le serveur où l'API écoute. Pour les récupérer, passer par les comptes `admin` et votre machine locale à l'aide de la commande `scp`.
    - lancer la commande suivante :

```
curl https://10.0.1.10:2376/images/json --key ./key.pem --cert
```

```
./cert.pem --insecure
```

    - Si tout est ok, vous devriez voir s'afficher des informations en JSON.
- Puis accéder à `https://manager.<domaine-sinp>` pour configurer cet instance (voir la doc dédiée).

From:
https://sinp-wiki.cbn-alpin.fr/ - **CBNA SINP**

Permanent link:
https://sinp-wiki.cbn-alpin.fr/serveurs/installation/web-srv/docker-api?rev=1671285159

Last update: **2022/12/17 13:52**